EXHIBIT 12

Appendix A

A Beginner's Guide to HTML

This is a primer for producing documents in HTML, the markup language used by the World Wide Web.

Introduction

Acronym expansion

WWW
World Wide Web

SGML
Standard Generalized Markup Language – This is perhaps best be thought of as a programming language for style sheets.

DTD
Document Type Definition – This is a specific implementation of document description using SGML. One way to think about this is: Fortran is to a computer program as SGML is to a DTD.

HTML
HyperText Markup Language – HTML is a SGML DTD. In practical terms, HTML is a collection of styles used to define the various components of a World Wide Web document.

What this primer doesn't cover

This primer assumes that you have:

- at least a passing knowledge of how to use NCSA Mosaic or other WWW browser
- a general understanding of how World Wide Web servers and client browsers work and
- access to a World Wide Web server for which you would now like to produce HTML documents

Creating HTML documents

HTML documents are in plain text format and can be created using any text editor (e.g., Emacs or vi on Unix machines). A couple of WWW browsers (tkWWW for X Window System machines and CERN's WWW browser for the NeXT) do include rudimentary HTML editors in a WYSIWYG environment, and you may want to try one of them first before delving into the details of HTML.

You can preview documents in progress with NCSA Mosaic (and some other WWW browers). Open the document using the Open Local option under the **File** *menu. Use the Filters, Directories, and Files fields to locate the document or enter the path and name of the document in the Name of Local Document to Open field. Press OK.*

[redacted] Appendix B

Mosaic for X version 2.0 Fill-Out Form Support

Here are details about what we have implemented for fill-out forms in Mosaic for X 2.0.

The FORM Tag

The FORM tag specifies a fill-out form within an HTML document. More than one fill-out form can be in a single document, but forms cannot be nested.

```
<FORM ACTION="url"> ...  </FORM>
```

The attributes are as follows:

- ACTION is the URL of the query server to which the form contents will be submitted; if this attribute is absent, then the current document URL will be used.
- METHOD is the HTTP/1.0 method used to submit the fill-out form to a query server. Which method you use depends on how your particular server works; we strongly recommend use of (or near-term migration to) POST. The valid choices are:
  - GET -- this is the default method and causes the fill-out form contents to be appended to the URL as if they were a normal query.
  - POST -- this method causes the fill-out form contents to be sent to the server in a data body rather than as part of the URL.
- ENCTYPE specifies the encoding for the fill-out form contents. This attribute only applies if METHOD is set to POST -- and even then, there is only one possible value (the default, `application/x-www-form-urlencoded`) so far.

NOTE: If you want to use the METHOD of type POST with the NCSA httpd you will need to get version 1.0a5 or later.

Inside a FORM you can have anything except another FORM. Specifically, INPUT, SELECT, and TEXTAREA tags are used to specify interface elements within the form.

Forms are not automatically visually differentiated from the rest of a document. We recommend using the HR (horizontal rule) tag before and after a form to cleanly differentiate it from surrounding text and/or other forms.

The INPUT Tag

Hypertext Transfer Prot[redacted] (HTTP)
Internet Draft
draft-ietf-iiir-http-00.txt

Tim Berners-Le[redacted] ERN
5 N[redacted]993
Expires 5 May 1994

Appendix C

Hypertext Transfer Protocol (HTTP)

A Stateless Search, Retrieve and Manipulation Protocol

Status of this memo

This document is an Internet Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are working documents valid for a maximum of six months. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress".

This document is a DRAFT specification of a protocol in use on the internet and to be proposed as an Internet standard. Discussion of this protocol takes place on the www-talk@info.cern.ch mailing list -- to subscribe mail to www-talk-request@info.cern.ch. Distribution of this memo is unlimited.

Abstract

HTTP is a protocol with the lightness and speed necessary for a distributed collaborative hypermedia information system. It is a generic stateless object-oriented protocol, which may be used for many similar tasks such as name servers, and distributed object-oriented systems, by extending the commands, or "methods", used. A feature if HTTP is the negotiation of data representation, allowing systems to be built independently of the development of new advanced representations.

Note: This specification

This HTTP protocol is an upgrade on the original protocol as implemented in the earliest WWW releases. It is back-compatible with that more limited protocol.

This specification includes the following parts:

The Request

Uniform Resource Locators
INTERNET DRAFT
IETF URL Working Group

Tim Berners-Lee
CERN
14 July 1993

Appendix D

Uniform Resource Locators

Status of this memo

Abstract

Many protocols and systems for document search and retrieval are currently in use, and many more protocols or refinements of existing protocols are to be expected in a field whose expansion is explosive.

These systems are aiming to achieve global search and readership of documents across differing computing platforms, and despite a plethora of protocols and data formats. As protocols evolve, gateways can allow global access to remain possible. As data formats evolve, format conversion programs can preserve global access. There is one area, however, in which it is impractical to make conversions, and that is in the names and addresses used to identify objects. This is because names and addresses of objects are passed on in so many ways, from the backs of envelopes to hypertext objects, and may have a long life.

This paper discusses the requirements on a universal syntax which can be used to refer to objects available using existing protocols, and may be extended with technology. It makes a recommendation for a generic syntax, and for specific forms for "Uniform Resource Locators" (URLs)of objects accessible using existing Internet protocols.

[redacted]